**Регрессия**

Основные цели этого задания:

* Научиться работать с документацией scikit-learn
* Научиться готовить данные к обучению модели и к ее валидации
* Научиться оценивать качество модели и выбирать лучшую модель, опираясь на метрики

***Задача:***

Построить модель машинного обучения для решения задачи регрессии: необходимо спрогнозировать прогресс заболевания диабетом через год после исходного уровня.

***План решения:***

1. В документации библиотеки scikit learn найдите, как загрузить датасет для построения модели прогноза прогрессирования заболевания через год после исходного уровня (load\_diabetes из sklearn.datasets).
2. Выведите содержимое поля DESCR, которое вернет функция, загружающая датасет, чтобы изучить содержимое датасета.
3. Подготовьте данные к обучению моделей: отделите целевой признак от датасета, разделите датасет на тренировочную и валидационную части.
4. Обучите решающее дерево и оцените адекватность обученной модели. Для этого достаточно сравнить метрики качества с метриками, получаемыми от простой модели. Например, если бы мы подавали на вход среднее значение целевой переменной на обучающей части выборки.
5. Обучите линейную регрессию и оцените ее адекватность.
6. Выберите лучшую модель и обоснуйте свой выбор.

**Классификация**

Построить модель машинного обучения, позволяющую предсказать, является ли цветок цветком класса Ирис-Вирджиния или нет.

***План решения:***

1. Из библиотеки scikit-learn подгрузите данные. Мы уже работали с ними в рамках Темы 1.
2. Сформируйте новое поле target\_virginica, которое будет равно из 0, если цветок не является классом 2, иначе  — 1. Убедитесь, что разметка выполнена корректно, рассчитав количество объектов каждого класса по новой целевой переменной target\_virginica.
3. Обучите библиотечную модель логистической регрессии на обучающей части данных.
4. Оцените качество обученной на предыдущем шаге модели, выведя матрицу ошибок для прогнозов обученной моделью на тестовой части данных.
5. Сконструируйте свой класс MyLogisticRegression, решающий задачу логистической регрессии методом градиентного спуска.
   1. Он должен инициализироваться величиной шага градиентного спуска и количеством итераций градиентного спуска.
   2. Должен содержать метод fit для обучения модели, на вход которой будут подаваться обучающие данные. Чтобы реализовать функцию sigmoid, можно воспользоваться библиотечной функцией np.exp. Для расчета градиента необходимо использовать формулу, приведенную в курсе в блоке «Логистическая регрессия для решения задачи бинарной классификации».
   3. Должен содержаться метод predict, который будет возвращать лейбл 0/1. Лейбл должен рассчитываться из вероятности принадлежности классу 1. Вероятность должна рассчитываться как сигмоида от результата произведения матрицы фичей X и вектора весов модели w.
6. Обучите модель с помощью реализованного класса с произвольными параметрами градиентного спуска.
7. Оцените качество модели, обученной с помощью класса MyLogisticRegression, выведя матрицу ошибок для прогнозов обученной моделью на тестовой части данных.
8. Подберите такие параметры градиентного спуска, чтобы разделение получилось не хуже, чем у библиотечной модели.